LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

LLNL-CONF-534497

High-Resolution Methods for Phase Space Problems in Complex Geometries

M. R. Dorr, P. Colella, J. A. Hittinger, D. F Martin, P. W. McCorquodale

March 1, 2012

Exascale Research Conference
Portland, OR, United States
April 16, 2012 through April 18, 2012

Disclaimer

High-Resolution Methods for Phase Space Problems in Complex Geometries

M. R. Dorr[1], P. Colella[2], J. A. F. Hittinger[1],
D. F. Martin[2], and P. W. McCorquodale[2]

[1]Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, 7000 East Avenue L-561, Livermore, CA 94550. *

[2]Applied Numerical Algorithms Group, Lawrence Berkeley National Laboratory, One Cyclotron Road Mail Stop 50A-1148, Berkeley, CA 94720. †

Abstract

We describe our progress in the development of fourth-order, mapped-multiblock, finite-volume discretizations applied to systems of partial differential equations posed in high-dimensional phase space. Our motivating application is the solution of gyrokinetic systems modeling edge plasma evolution in tokamak fusion reactors. For these problems, the use of computational coordinates aligned with magnetic field lines is highly advantageous in accommodating strong anisotropy, except near the X point in axisymmetric simulations where such a coordinate system has a singularity. We summarize our approach for addressing this issue in the development of our COGENT testbed edge code as well as future plans.

1 Introduction

Many problems in mathematical physics are posed in phase space coordinates, typically position and momentum. The need to numerically solve partial differential equations in 4, 5 or even 6 (*e.g.*, 3 spatial and 3 momentum) dimensions plus time requires efficient discretizations and advanced implementation strategies. Even then, high-performance computing resources are essential to access the degree of resolution required by scientific applications.

We have therefore been investigating numerical discretization and implementation strategies applicable to phase space problems on petascale systems and beyond. Central to our approach is the use of high-order (fourth-order or better), finite-volume discretizations on block structured grids. It has long been recognized that high-order methods enable more efficient discretizations measured in terms of accuracy per degree of freedom. Exploiting this fact to address the high dimensionality of phase space problems is therefore essential. More recently, interest in high-order methods has increased due to the ascendence of multi- and many-core processor

*This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. LLNL-CONF-534497.

†Research supported by the Office of Advanced Scientific Computing Research of the US Department of Energy under contract number DE-AC02-05CH11231.

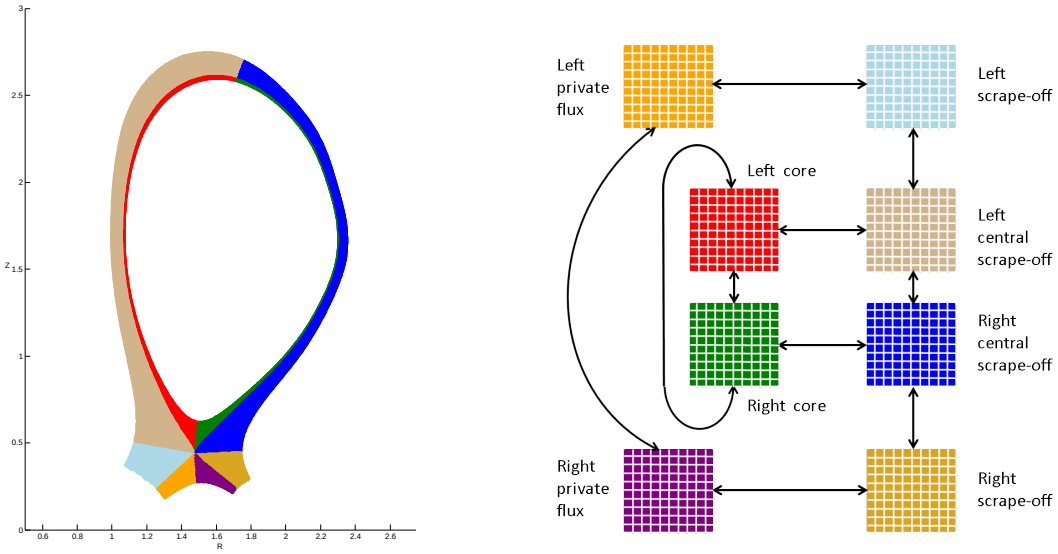


Figure 1: Plasma edge geometry (left) and multiblock, locally rectangular computational domain (right). Arrows indicate the inter-block connectivity.

architectures. Generally speaking, high-order methods possess greater arithmetic density than low-order methods. With their regular and more easily predictable data access patterns, the use of block-structured grids is also well-aligned with architectural trends, as has already been demonstrated with present-day cache-based designs.

The applicability of high-order, block-structured finite volume discretizations is further enhanced when combined with coordinate mapping. A particularly interesting example is provided by the specific application driving our algorithm development, which is the solution of gyrokinetic systems describing the edge plasma of tokamak fusion reactors. The relative size of important length scales dictates a kinetic model, where each plasma species is described by a distribution function in phase space coordinates. We consider gyrokinetic systems to have the general conservative form

$$\partial_t f + \nabla \cdot [u(f) f] = C(f), \tag{1}$$

where $f \equiv f(x, v, t)$ is a distribution function of the phase space coordinates (x, v) and C denotes a collision operator. The advection velocity u depends upon the distribution function in a nonlinear and highly nonlocal manner involving the solution of some form of Maxwell's equations. Further details can be found in [4]. In addition to the high-dimensionality challenge, the edge problem is posed in a unique geometry defined by magnetic flux surfaces (Figure 1 (left)). Strong anisotropy along versus transverse to magnetic field lines motivates the use of coordinates aligned with flux surfaces. Such coordinates can be defined by mapping logically distinct regions of the edge geometry (left/right core, left/right central scrape-off, left/right scrape-off, left/right private flux) to a multiblock computational domain (Figure 1 (right)), where each block is gridded uniformly.

2 Approach

We summarize here our general approach for the systematic development of high-order finite volume discretizations in mapped mulitblock coordinates.

2.1 Single block

Suppose that we have a smooth mapping $\boldsymbol{X} = \boldsymbol{X}(\boldsymbol{\xi})$, $\boldsymbol{X} : [0,1]^D \to \Omega$ from the unit cube onto the spatial domain $\Omega \in \mathbb{R}^D$. Given this mapping, the divergence of a vector field on Ω can be written in terms of derivatives in $[0,1]^D$, which serves as our computational domain. That is,

$$\boldsymbol{\nabla}_{\boldsymbol{x}} \cdot \boldsymbol{F} = \frac{1}{J} \boldsymbol{\nabla}_{\boldsymbol{\xi}} \cdot (\boldsymbol{N}^T \boldsymbol{F}), \tag{2}$$

$$J = \det(\boldsymbol{\nabla}_{\boldsymbol{\xi}} \boldsymbol{X}) \ , \ \left(\mathbf{N}^T\right)_{p,q} = \det\left(\mathbf{R}_p(\boldsymbol{\nabla}_{\boldsymbol{\xi}} \mathbf{X}, \mathbf{e}^q)\right), \tag{3}$$

where $\mathbf{R}_p(\mathbf{A}, \mathbf{v})$ denotes the matrix obtained by replacing the p^{th} row of the matrix $\mathbf{A}$ by the vector $\mathbf{v}$, and $\mathbf{e}^d$ denotes the unit vector in the d^{th} coordinate direction. The relationship (2) is an easy consequence of the chain rule, equality of mixed partials, and Cramer's rule.

In a finite volume approach, Ω is discretized as a union of control volumes. When using mapped coordinates, we define control volumes in Ω as the images $\boldsymbol{X}(V_{\boldsymbol{i}})$ of the cubic control volumes $V_{\boldsymbol{i}} \subset [0,1]^D$. Then, by changing variables and applying the divergence theorem, we obtain the flux divergence integral over a physical control volume $\boldsymbol{X}(V_{\boldsymbol{i}})$ by

$$\int\limits_{\boldsymbol{X}(V_{\boldsymbol{i}})} \boldsymbol{\nabla}_{\boldsymbol{x}} \cdot \boldsymbol{F} d\boldsymbol{x} = \int\limits_{V_{\boldsymbol{i}}} \boldsymbol{\nabla}_{\boldsymbol{\xi}} \cdot (\boldsymbol{N}^T \boldsymbol{F}) d\boldsymbol{\xi} = \sum_{\pm=+,-} \sum_{d=1}^{D} \pm \int\limits_{A_d^{\pm}} (\boldsymbol{N}^T \boldsymbol{F})_d dA_{\boldsymbol{\xi}}, \tag{4}$$

where the $A_d^{\pm}$ are upper and lower faces of cell $V_{\boldsymbol{i}}$ in the d-th direction. As described in [2], the integrals on the cell faces $A_d^{\pm}$ can be approximated using a formula for the average of a product in terms of fourth-order accurate face averages of each factor. This yields

$$\int\limits_{\boldsymbol{X}(V_{\boldsymbol{i}})} \boldsymbol{\nabla}_{\boldsymbol{x}} \cdot \boldsymbol{F} d\boldsymbol{x} = h^2 \sum_{d=1}^{D} \sum_{\pm=+,-} \pm F^d_{\boldsymbol{i} \pm \frac{1}{2}\boldsymbol{e}^d} + O(h^4), \tag{5}$$

where

$$F^d_{\boldsymbol{i} \pm \frac{1}{2}\boldsymbol{e}^d} \equiv \sum_{s=1}^{D} \langle N_d^s \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} \langle F^s \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d} + \frac{h^2}{12} \sum_{s=1}^{D} \left(\boldsymbol{G}_0^{\perp,d} \langle N_d^s \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}) \right) \cdot \left(\boldsymbol{G}_0^{\perp,d} (\langle F^s \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}) \right), \tag{6}$$

Here, $\boldsymbol{G}_0^{\perp,d}$ is the second-order accurate central difference approximation to the component of the gradient operator orthogonal to the d-th direction: $\boldsymbol{G}_0^{\perp,d} \approx \boldsymbol{\nabla}_{\boldsymbol{\xi}} - \boldsymbol{e}^d \frac{\partial}{\partial \xi_d}$, and the operator $\langle \cdot \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$ denotes a fourth-order accurate average over the face centered at $\boldsymbol{i} + \frac{1}{2}\boldsymbol{e}^d$. F^s is the s-th component of $\boldsymbol{F}$ and N_d^s is the (s,d)-th element of the matrix $\boldsymbol{N}$. To reduce the stencil size, alternative expressions can be obtained by replacing the averages $\langle f \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$ and/or $\langle g \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$ used in the transverse gradients $\boldsymbol{G}_0^{\perp,d}$ by the corresponding face-centered pointwise values $f_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$ and/or $g_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$, respectively.

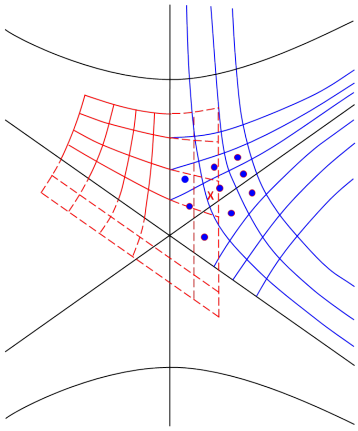


Figure 2: Interpolation of ghost cells on smoothly-extended grid. The red X indicates the center of a ghost control volume for which a value is to be interpolated, while the blue dots indicate the centers of nearby control volumes that will be used in the least-squares calculation.

In [2], we demonstrated that the computation of the face averages $\langle N_d^s \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}^d}$ can be reduced to integrals over cell edges. Moreover, the freestream property that the divergence of a constant vector field computed by (6) is zero to machine accuracy also holds.

2.2 Multiblock

To extend the methods just described to multiblock grids, we apply the various stencil operations by computing ghost-cell values on grids that are smooth extensions of the original block or union of rectangles at a level (Figure 2), together with a mechanism for obtaining single-valued fluxes at block boundaries. To compute these ghost-cell values, we use a least-squares approach that allows us to obtain high-order accuracy independent of the degree of smoothness of the mesh. We compute a polynomial interpolant in the neighborhood of a ghost cell of the form

$$\varphi(\boldsymbol{x}) \approx \sum_{p_d \geq 0; p_1 + \cdots + p_D \leq P-1} a_{\boldsymbol{p}} \boldsymbol{x}^{\boldsymbol{p}} \ , \ \boldsymbol{p} = (p_1, \ldots, p_D) \ , \ \boldsymbol{x}^{\boldsymbol{p}} = x_1^{p_1} \ldots x_D^{p_D}. \tag{7}$$

We will assume that we know the conserved quantities in a collection of control volumes $\boldsymbol{v} \in \mathcal{V}$. In that case,we impose the conditions

$$\int_{\boldsymbol{v}} \varphi(\boldsymbol{\xi}) d\boldsymbol{\xi} = \sum_{p_d \geq 0; p_1 + \cdots + p_D \leq P-1} a_{\boldsymbol{p}} \int_{\boldsymbol{v}} \boldsymbol{x}(\boldsymbol{\xi})^{\boldsymbol{p}} d\boldsymbol{\xi} \ , \ \boldsymbol{v} \in \mathcal{V}. \tag{8}$$

The integrals on the left-hand side can be computed to any order from the known integrals of the conserved quantities $J\phi$, and the integrals of $\boldsymbol{x}^{\boldsymbol{p}}$ can be computed directly from the grid mapping. Thus, this constitutes a system of linear equations for the interpolation coefficients $a_{\boldsymbol{p}}$. Generally, we choose the number of equations to be greater than the number of unknowns in such a way that the resulting overdetermined system has maximal rank, so that it can be solved using least squares. In the case where we are computing an interpolant onto a finer grid from a coarser one in a locally-refined mesh calculation, we impose the conservation condition as a linear constraint.

3 Accomplishments

3.1 COGENT algorithm testbed

To test the performance of our algorithms on gyrokinetic edge systems, we have developed a testbed code named COGENT (COntinuum Gyrokinetic Edge New Technology). COGENT is a 4D (2 configuration space plus 2 velocity space coordinates) continuum gyrokinetic edge code under development by the ESL ASCR/FES collaboration. COGENT solves the "full-f" gyrokinetic Vlasov-Poisson system presented in [5] expressed in conservation form. An arbitrary number of kinetic species can be simulated. A Boltzmann electron model provides a fast-running option for studying ion dynamics (including prefactor options to, *e.g.*, maintain charge neutrality on closed flux surfaces). The gyrokinetic system is spatially discretized using the fourth-order finite-volume scheme in a mapped coordinate system summarized above.

The semi-discrete system is integrated temporally using a fourth-order, explicit Runge-Kutta (RK4) scheme. At each stage of the RK4 algorithm, the gyrokinetic Poisson equation is solved using the RK4-predicted distribution functions, yielding the electric field needed to compute the phase space velocities. When the Boltzmann electron option is selected, the resulting nonlinear Poisson-Boltzmann equation is solved using a Newton iteration. In each Newton iteration, the nonsymmetric Jacobian system is solved using BiCGStab (Bi-Conjugate Gradient Stabilized) with a preconditioner that invokes a Hypre multigrid solver. The nonsymmetry of the Jacobian is a consequence of the form of the prefactor in the Boltzmann relation.

COGENT is built upon the Chombo library [3] under development by the SciDAC-3 FASTMath institute to facilitate the creation of block-structured adaptive mesh refinement (AMR) applications. Although COGENT does not currently utilize Chombo's AMR capabilities, a future development path is nevertheless provided. Chombo provides support for the mapped-grid, finite-volume formalism described above. This includes the construction of discrete metric quantities from a user-specified mapping and the computation of fourth-order face-averaged fluxes. COGENT utilizes Chombo's data containers for mesh-dependent quantities distributed over processors. Such quantities are functions of configuration space (*e.g.*, potential) or phase space (*e.g.*, distribution functions), each of which can be domain decomposed independently. Injection and projection operators between configuration/velocity and phase space have also been developed in COGENT.

3.2 High-order discretization near X points

As described above, the primary motivation for the use of mapped coordinates in the discretization of plasma edge problems is to accommodate strong anisotropy along magnetic field lines. Although field lines always define smooth curves in full three-dimensional space, for axisymmetric simulations in which the toroidal component is projected onto a poloidal slice, a difficulty arises near the X point, where the poloidal field component vanishes. As seen in the left plot in Figure 3, field line projections (the black dashed lines) become increasingly "kinked" approaching the X point, where the projections degenerate to a single point. It is therefore not possible to define a smooth field-aligned coordinate system in this region. However, because the poloidal component of the magnetic field vanishes at the X point and is small in a neighborhood, field-induced anisotropy is not a concern, and field aligned coordinates are no longer necessary. To apply our high-order, mapped multiblock discretization strategy in the vicinity

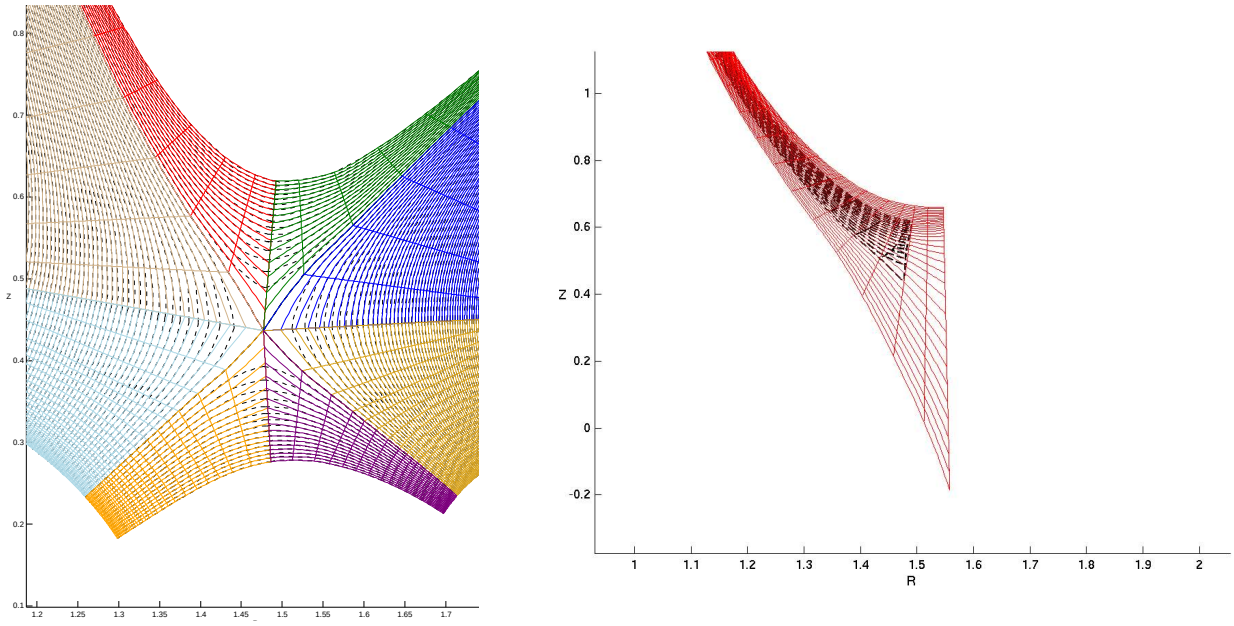


Figure 3: Left: Multiblock grid and field lines (dashed lines) near the X point. Right: Field line following grid (blue) and smooth extension of the modified grid (red) through the X point.

of X points, we must therefore transition from a field line following coordinate system to one that can be smoothly extended through the X point.

The magnetic geometry is determined by an equilibrium model or magnetohydrodynamic (MHD) calculation. In the approach described here, we assume that we are provided with geometry data in terms of a table describing the (R, Z) coordinates of points along field lines in each block, and the corresponding field data. The variation of the coordinates between the specified points is undefined, but is assumed to be smooth. In COGENT, an arbitrary order B-spline interpolant is used to generate a smooth mapping from the input coordinate table, following a preprocessing step in which the table points near the X point are redistributed and additional points are added along all block boundaries to provide a smoothly extended mapping.

The preprocessing step begins with the specification of a neighborhood of radius $\mathcal{R}$ about the X point beyond which the coordinate system must remain field aligned. $\mathcal{R}$ can be determined from the magnitude of the poloidal field component or some other physical quantities of interest related to the anisotropy concern. Within the $\mathcal{R}$-neighborhood, we identify the intersection of field lines with block boundaries. We then redistribute the intersection points along the block boundaries using a quartic polynomial mapping whose coefficients are chosen so as to ensure a smooth transition at radius $\mathcal{R}$ (by matching derivatives) to a more equally spaced distribution near the X point. The next step is to redefine the radial grid lines intersecting the $\mathcal{R}$-neighborhood such that they are straight and their intersection with block boundaries containing the X point are equidistributed. Although this step moves grid points outside of the $\mathcal{R}$-neighborhood of the X point, such points are always moved along field lines using a

root finding algorithm, so that field lines are preserved. The goal is to construct in the $\mathcal{R}$-neighborhood a grid that is as close to rectilinear as possible, since the final step is to extend the grid through the X point using an extrapolation with monotone splines (implemented using the `pchip` interpolator in the Matlab script we have developed to perform the grid preprocessing). Figure 3 (left) shows the resulting modified grid in each of the blocks near the X point. The dashed black curves indicate the field lines, which remain aligned with the grid away from the X point. On the right in Figure 3 is a comparison of the original grid (blue) in the left core block and the modified/extended grid (red). Similar extensions through the X point are obtained for the other seven blocks.

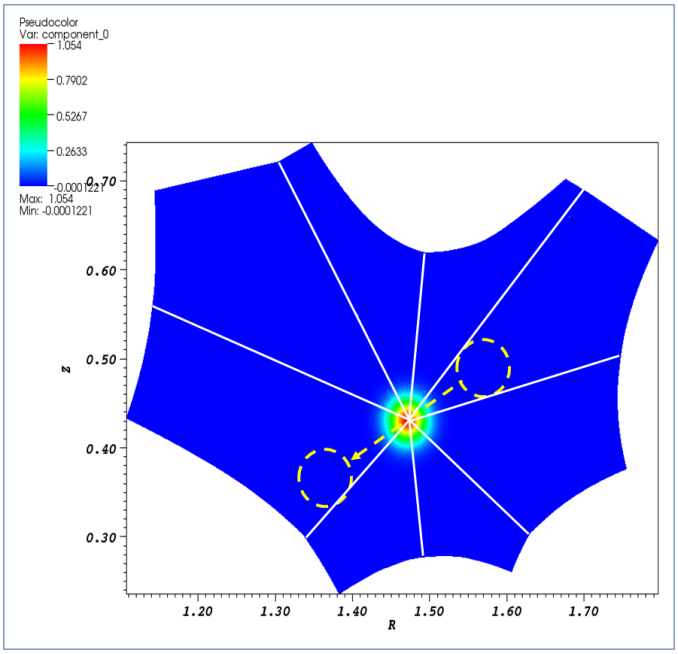
3.3 Example

To test the accuracy of our mapped-multiblock, finite-volume discretization near the X point of a plasma edge geometry, we considered the linear advection of a Gaussian "blob". Because the rest of the edge domain was irrelevant to this test, the domain was truncated to include just a region about the X point. As indicated by the dashed yellow lines in Figure 4, the blob was initialized slightly above and to the right of the X point, then advected downward and to the left, passing directly through the X point. The test was performed to the same ending time on four grids generated by successive refinement by a factor of two. Richardson extrapolation was then applied to estimate the convergence rates, which are also shown in the table in Figure 4. As the grid is refined, the convergence rate appears to be tending toward fourth-order, but then decreases at the finest resolution. We believe that this behavior is due to a loss of accuracy resulting from ill-conditioning of the normal equations used to solve the least squares system for the multiblock interpolation coefficients. We are currently implementing an improved algorithm for selecting the interpolation stencil at the X point, for which preliminary tests have shown considerably improved condition numbers. We are also replacing the normal equation solution of the least squares system (7) (which tends to square the condition number) by a QR factorization.

4 Future Work

In addition to the work to improve the condition number of block boundary interpolation procedures, in the near-term, we are assembling the components necessary to simulate plasma flow on realistic tokamak edge geometries, as shown in Figure 1. This includes the implementation of more general, fourth-order characteristic boundary conditions suitable for artificial boundaries and diverter plates. Once in realistic geometries, we will be able to investigate more systematically the optimal choice of the $\mathcal{R}$-neighborhood in which the mapping diverges from flux surfaces. Demonstrating fourth-order convergence of to solutions of the gyrokinetic Vlasov-Poisson in a genuine edge geometry will be a major accomplishment.

Beyond this milestone, we will pursue three main issues: time integration, error estimation, and optimization. A wide range of timescales in physically interesting problems will necessitate more sophisticated time integration procedures; we will investigate the suitability of several implicit-explicit partitioned time integrators. The ability to estimate efficiently the resulting discretization error in computed solutions is important not only to understand the minimum resolution requirements for expensive phase-space calculations, but also for uncertainty quan-



4h/2h 2h/h	Extrapolated convergence rate
512/1024 1024/2048	3.56 (L1) 3.45 (L2) 2.79 (Max)
1024/2048 2048/4096	3.77 (L1) 3.72 (L2) 3.77 (Max)
2048/4096 4096/8192	2.77 (L1) 2.74 (L2) 3.22 (Max)

Figure 4: Blob propagation near the X point, grid refinement pairs, and Richardson extrapolated converge rates for the blob propagation test.

tification. We have begun work on a nonlinear error transport (*e.g.*, [1]) capability to provide for such estimates. Finally, working with other researchers in the SciDAC-3 FASTMath Institute, we will study the performance of the mapped multiblock capabilities in Chombo within the context of our phase-space application and develop more efficient algorithms and implementations.

References

[1] J. W. Banks, J. A. F. Hittinger, J. M. Connors, and C. S. Woodward. Numerical error estimation for nonlinear hyperbolic PDEs via nonlinear error transport. *Comput. Methods Appl. Mech. Eng.*, 213–216:1–15, 2012.

[2] P. Colella, M. R. Dorr, J. A. F. Hittinger, and D. F. Martin. High-order, finite-volume methods in mapped coordinates. *J. Comput. Phys.*, 230:2952–2976, 2011. Also available as Lawrence Livermore National Laboratory report LLNL-JRNL-385708.

[3] P. Colella, D. T. Graves, T. J. Ligocki, D. F. Martin, D. Modiano, D. B. Serafini, and B. Van Straalen. Chombo Software Package for AMR Applications - Design Document. `http://seesar.lbl.gov/anag/chombo`.

[4] M. R. Dorr, R. H. Cohen, P. Colella, M. A. Dorf, J. A. F. Hittinger, and D. F. Martin. Numerical simulation of phase space advection in gyrokinetic models of fusion plasmas. In *Proceedings of SciDAC 2010, Chattanooga, TN*, July 2010.

[5] T. S. Hahm. Nonlinear gyrokinetic equations for turbulence in core transport barriers. *Phys. Plasmas*, 3(12):4658–4664, 1996.